Computers and Fluids 144 (2017) 34–43

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

An efficient unified iterative scheme for moving boundaries in lattice Boltzmann method

Junjie Hu, Shi Tao, Zhaoli Guo*

State Key Laboratory of Coal Combustion, School of Energy and Power Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

ARTICLE INFO

Article history:
Received 29 June 2016
Revised 16 October 2016
Accepted 9 December 2016
Available online 13 December 2016

Keywords:
Particulate flows
Curved boundary condition
Refilling algorithm
Lattice Boltzmann method

ABSTRACT

The lattice Boltzmann equation (LBE) is an efficient kinetic method for particulate flows. Two key issues should be addressed in the implementation of LBE for such systems, i.e., how to treat the curved surface of a solid particle on a uniform Cartesian grid, and how to initialize the state of a fresh node coming from the moving particle. These two key issues are usually considered separately in previous studies. In this work, we propose an efficient unified iterative scheme (UIS) to treat both the issues simultaneously. On one hand, the present method provides a consistent treatment for both the boundary nodes and fresh nodes, on the other hand, to enforce the no-slip boundary condition and decrease the inconsistency between the constructed distribution functions and those evolutionary ones, an enforced iteration (EI) is employed. To describe the inconsistency quantitatively, the inconsistency degree is defined. Simulations of several typical problems are conducted, and the numerical accuracy, computational efficiency and ability to treat moving boundaries are validated. Compared with the combination method, the inconsistency degree around the moving body and spurious force fluctuation are suppressed significantly due to the improved consistency.

1. Introduction

Particulate flows are of great importance in numerous industrial and natural systems, such as fluidized beds, lubricated transport, fluvial erosion, river sediment resuspension and sand storms [1,2]. As a promising numerical technique in the computational fluid dynamics (CFD), the lattice Boltzmann method (LBM), which is based on the evolution of discrete distribution functions of the fluid molecules, has proven to be an efficient and robust method for simulating particulate flows in the past two decades [3–13]. In LBM, the fluid-solid interaction is represented via the boundary condition on the solid surface, and therefore it is critical to set up an accurate rule to realize the boundary condition in terms of the distribution functions [14,15]. Another key point for moving boundaries is how to treat the fresh fluid nodes that move into the fluid from the solid body due to the movement of the solid body. To construct the unknown distribution functions at the fresh nodes, a refilling algorithm is needed. These two crucial issues have been studied widely in previous studies [12,13,16–26].

For the first key problem, i.e., realizing the boundary condition on the solid surface, Ladd made the first attempt by representing the surface with a set of mid-points of the links across the solid surface, on which the bounce-back rule is imposed to realize the no-slip boundary condition [3,4]. It is clear that in Ladd's method the physical surface of the particle is approximated by a zigzag shape [2,20]. As pointed out in [7,27], such a treatment leads to a Knudsen layer near the particle surface and the particle size should be rescaled. Consequently, the overall accuracy of Ladd's method is of first-order. To preserve the geometric integrity and improve the computational accuracy, a number of curved boundary conditions have been developed [14–21]. For instance, Filippova et al. [16] presented a boundary condition which provided a second-order accurate treatment for curved boundaries. However, the numerical stability of their method is far from satisfactory. To improve the numerical stability, Mei et al. [17] presented an improved version. Bouzidi et al. [18] proposed a different but simpler boundary condition for curved boundaries based on the interpolation bounce-back scheme, while Guo et al. [19] developed a non-equilibrium extrapolation (NEE) method, in which the distribution functions at a solid node are decomposed into their equilibrium and non-equilibrium parts. Recently, Yin et al. [14] proposed an improved bounce-back scheme, in which the velocity at the mid-point of the link is used, instead of the boundary velocity in Ladd's method. For the above curved boundary conditions, different interpolation schemes are employed based on the fraction of the intersected link in the fluid region, which may induce some unphysical oscillations near the solid surface. To

* Corresponding author.
E-mail address: zlguo@hust.edu.cn (Z. Guo).

http://dx.doi.org/10.1016/j.compfluid.2016.12.007
0045-7930/

overcome the abrupt discontinuity in the boundary condition, Yu et al. [20] proposed a unified interpolation bounce-back scheme, which is applicable in the whole range with improved numerical stability. Although the boundary conditions based on the interpolation bounce-back scheme are of second-order accuracy, the relative errors are claimed to be viscosity-dependent [21,28]. To conquer the drawback, Ginzburg et al. [21] proposed a viscosity-independent multireflection boundary condition.

For the second key problem, i.e., the refilling of fresh nodes coming from the moving solid body, a number of algorithms have been developed [12,13]. Luo et al. [22] presented a second-order extrapolation method along a specified direction to initialize the fresh node. Li et al. [23] extrapolated the unknown distribution functions at the fresh node with those averaged ones at its surrounding fluid nodes. Mei et al. [24] introduced a consistent initialization method, in which the density field is constructed related to a known velocity field. Later, Caiazzo et al. [25] proposed a simple refilling algorithm based on the non-equilibrium extrapolation method. Recently, Chen et al. [26] presented a local evolutionary iteration method, in which the local collision and streaming are conducted to update the distribution functions, density and velocity at the fresh node.

For moving boundaries, boundary conditions and refilling algorithms are recognized as different topics in previous studies. However, the essences of these two seemingly different problems are the same, that is, how to specify the unknown distribution functions at certain nodes near the solid surface. The previous boundary conditions and refilling algorithms constructed from a separate manner may induce some inconsistency. Therefore, a consistent treatment of these two key problems is desirable, which is the aim of the present work.

The rest of this paper is organized as follows. Section 2 briefly describes the lattice Boltzmann method, and Section 3 presents the consistent method for moving boundaries. In Section 4, the numerical accuracy, computational efficiency and ability to treat moving boundaries of the present method are validated by several typical problems, and Section 5 concludes the paper.

2. Lattice Boltzmann method

Instead of solving the Navier–Stokes (NS) equations directly, LBM is a mesoscopic method derived from the microscopic Boltzmann equation, whose evolution equation is

$$f_i(\boldsymbol{x}+\boldsymbol{c}_i\delta t, t+\delta t) - f_i(\boldsymbol{x}, t) = \Omega_i(f), \tag{1}$$

where $f_i(\boldsymbol{x}, t)$ is the distribution function at position $\boldsymbol{x}$ and time t for particles with velocity $\boldsymbol{c}_i$ along the ith direction of the lattice, $\Omega_i(f)$ is the discrete collision operator, and δt is the time step. In practice, the evolution of the distribution functions is decomposed into two steps, i.e., collision step

$$f_i'(\boldsymbol{x}, t) = f_i(\boldsymbol{x}, t) + \Omega_i(f), \tag{2}$$

and streaming step

$$f_i(\boldsymbol{x}+\boldsymbol{c}_i\delta t, t+\delta t) = f_i'(\boldsymbol{x}, t), \tag{3}$$

where $f_i'(\boldsymbol{x}, t)$ is the post-collision distribution function.

Due to its superior numerical stability [29,30], the multi-relaxation-time (MRT) model is employed in this work, in which the collision operator is

$$\Omega_i(f) = -(M^{-1}SM)_{ij}(f_j - f_j^{eq}), \tag{4}$$

where M is the transform matrix, S is the diagonal relaxation matrix, and f_i^{eq} is the equilibrium distribution function (EDF), which is a function of the macroscopic quantities. Note that the Einstein summation notation is adopted here.

In the present study, the D2Q9 (2-dimension and 9-velocity) model [31] is adopted, in which the transform matrix M is

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{pmatrix}, \tag{5}$$

and the relaxation matrix S is

$$S = diag(\tau_\rho, \tau_e, \tau_\epsilon, \tau_j, \tau_q, \tau_j, \tau_q, \tau_s, \tau_s)^{-1}, \tag{6}$$

where τ_ρ and τ_j are related to the conserved moments, while the other relaxation times are related to the non-conserved moments. The EDF is defined as

$$f_i^{eq}(\rho, \boldsymbol{u}) = w_i\rho\left[1 + \frac{\boldsymbol{c}_i\cdot\boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_i\cdot\boldsymbol{u})^2}{2c_s^4} - \frac{\boldsymbol{u}\cdot\boldsymbol{u}}{2c_s^2}\right], \tag{7}$$

where ρ is the fluid density, $\boldsymbol{u}$ is the fluid velocity, w_i is the weight parameter given by

$$w_i = \begin{cases} 4/9 & \text{for } i = 0, \\ 1/9 & \text{for } i = 1, 2, 3, 4, \\ 1/36 & \text{for } i = 5, 6, 7, 8, \end{cases} \tag{8}$$

and the particle velocity $\boldsymbol{c}_i$ is

$$\boldsymbol{c}_i = \begin{cases} (0, 0) & \text{for } i = 0, \\ c(\cos[(i-1)\pi/2], \sin[(i-1)\pi/2]) & \text{for } i = 1, 2, 3, 4, \\ \sqrt{2}c(\cos[(2i-1)\pi/4], \sin[(2i-1)\pi/4]) & \text{for } i = 5, 6, 7, 8, \end{cases} \tag{9}$$

where $c = \delta x/\delta t$ is the lattice speed, δx is the grid spacing, and $c_s = c/\sqrt{3}$ is the speed of sound. The fluid density ρ and velocity $\boldsymbol{u}$ are obtained by

$$\rho = \sum_i f_i, \quad \rho\boldsymbol{u} = \sum_i \boldsymbol{c}_i f_i. \tag{10}$$

The fluid pressure is defined as $p = \rho c_s^2$, and the kinematic viscosity is related to the relaxation time τ_s as $\nu = c_s^2(\tau_s - 1/2)\delta t$.

3. The unified iterative scheme for moving boundaries

As sketched in Fig. 1, the lattice nodes are classified into two categories by the solid surface Γ, i.e., fluid nodes and solid nodes. The former is occupied by the fluid while the latter is covered by the solid body. Due to the movement of the solid body, the fluid nodes near the solid surface are further classified into two types, namely, boundary nodes that are occupied by the fluid at both the present time t and previous time step $t - \delta t$, and fresh nodes that were covered by the solid body at time $t - \delta t$. Generally, at a boundary node, the distribution functions with links to other fluid nodes are obtained after the streaming step, but those with links to solid nodes need to be specified according to certain kinetic boundary conditions; unlike boundary nodes, the distribution functions at a fresh node are all unknown when it moves into the fluid from the solid body and should be initialized with certain refilling algorithms. In previous studies [12,13], the unknown distribution functions at boundary nodes and fresh nodes are treated with separate rules, which may lead to some inconsistency. In the present study, we propose a consistent method to specify the unknown distribution functions at both types of nodes.

After the collision step and streaming step, the distribution functions at fluid nodes and boundary nodes with links to other